

ALGORITMOS DE BUSQUEDA

Pamela Alessandra, Martínez Villegas, 20230067|

Escuela de Mecánica Eléctrica, Facultad de Ingeniería, Universidad de San Carlos de Guatemala

Resumen—Estas instrucciones son una guía básica para la preparación de un trabajo para ser presentado ante los cursos 0769 Introducción a la Programación de Computadoras y 0980 Proyectos de Computación Aplicados a Ingeniería Electrónica. Este documento es en sí mismo un ejemplo del formato deseado (inclusive este Resumen) y **puede ser usado como una plantilla**.

El Resumen está limitado a 150 palabras y no debe contener ecuaciones, figuras, tablas, o referencias. Debe concisamente enunciar que fue hecho, como fue hecho, resultados principales, y su trascendencia.

I. MARCO TEÓRICO

A. Figura y tablas

```

C SECUENCIAL.c 1
C SECUENCIAL.c > ...
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<conio.h>
4 int BusquedaSecuencial(int *parray, int buscar, int nelem);
5 void main(){
6     int *pdatos, nelem, d, buscar, result;
7     printf("Cuantos elementos desea en el array?");
8     scanf("%d", &nelem);
9     pdatos = (int*) malloc(nelem*sizeof(int));
10    if(pdatos == NULL){
11        printf("Insuficiente Espacio de Memoria");
12        exit(-1);
13    }
14    for (d = 0; d < nelem; d++){
15        printf("Elemento[%d]:", d);
16        scanf("%d", (pdatos+d));
17    }
18    printf("\n Que elemento desea buscar?");
19    scanf("%d", &dbuscar);
20
21    result = BusquedaSecuencial(pdatos, dbuscar, nelem);
22
23    if(result != -1)
24        printf("\n %d Se encuentra en la posicion %d del array.\n", dbuscar, result);
25    else
26        printf("\n %d No se encontro en el array.\n", dbuscar);
27 }
28 int BusquedaSecuencial(int *parray, int buscar, int elem){
29     int i;
30
31     for(i = 0; i<elem; i++){
32         if(*(parray+i) == buscar)
33             return(i);
34     }
35     return(-1);
36 }

```

Fig1.Código secuencial completo.

```

PS C:\Users\HP\Desktop\Pame\output> & .\'SECUENCIAL.exe'
Cuantos elementos desea en el array?6
Elemento[0]:1
Elemento[1]:2
Elemento[2]:3
Elemento[3]:4
Elemento[4]:5
Elemento[5]:6

Que elemento desea buscar?1

1 Se encuentra en la posicion 0 del array.
PS C:\Users\HP\Desktop\Pame\output>

```

Fig2. Funcionamiento del Código secuencial.

```

C BINARIO.c X
C BINARIO.c > ...
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<conio.h>
4 int BusquedaBinaria(int datos[], int nelem, int clave);
5 void main(){
6     int *pdatos, nelem, dbuscar, d, result;
7     printf("Cuantos elementos desea en el array?");
8     scanf("%d", &nelem);
9     pdatos = (int*) malloc(nelem*sizeof(int));
10    if(pdatos==NULL){
11        printf("Insuficiente espacio de memoria");
12        exit(-1);
13    }
14    for (d = 0; d < nelem; d++){
15        printf("Elemento[%d]:", d);
16        scanf("%d", (pdatos+d));
17    }
18    printf("\n Que elemento desea buscar?");
19    scanf("%d", &dbuscar);
20
21    result = BusquedaBinaria(pdatos, nelem, dbuscar);
22
23    if(result!=-1)
24        printf("\n %d Se encuentra en la posicion %d del array.", dbuscar, result);
25    else
26        printf("\n %d no se enecntro en el array.\n", dbuscar);
27 }
28 int BusquedaBinaria(int lista[], int n, int clave){
29     int central, bajo, alto;
30     int valorCentral;
31     bajo = 0;
32     alto = n-1;
33     while (bajo <= alto){
34         central = (bajo+alto)/2;
35         valorCentral = lista[central];
36         // printf("\n\n %d el valor central actual es.\n\n", central);
37
38         if(clave == valorCentral)
39             return(central);
40         else if (clave < valorCentral)
41             alto = central-1;
42         else
43             bajo = central+1;
44     }
45     return(-1);
46 }

```

Fig3. Código binario completo.

```

PS C:\Users\HP\Desktop\Pame\output> & .\BINARIO.exe
Cuantos elementos desea en el array?5
Elemento[0]:1
Elemento[1]:2
Elemento[2]:3
Elemento[3]:4
Elemento[4]:5

Que elemento desea buscar?4

4 Se encuentra en la posición 3 del array.
PS C:\Users\HP\Desktop\Pame\output> Z

```

Fig4. Funcionamiento del código binario.

```

C BINARIO2.c X
C BINARIO2.c > ...
1  #include<stdio.h>
2  #include<stdlib.h>
3  #include<conio.h>
4  int BusquedaBinaria(int datos[], int nelem, int clave);
5  void OrdenarBurbuja(int datos[], int nelem){
6      int i, j, temp;
7      for(i=0; i<nelem; i++){
8          for(j=0; j<(nelem-1); j++){
9              if(datos[j]> datos[j+1]){
10                 temp = datos[j];
11                 datos[j] = datos[j+1];
12                 datos[j+1] = temp;
13             }
14         }
15     }
16     void main(){
17         int *pdatos, nelem, dbuscar, d, result;
18         printf("Cuantos elementos desea en el array?");
19         scanf("%d", &nelem);
20         pdatos = (int*) malloc(nelem*sizeof(int));
21         if(pdatos==NULL){
22             printf("Insuficiente espacio de memoria");
23             exit(-1);
24         }
25         for (d = 0; d < nelem; d++){
26             printf("Elemento[%d]:", d);
27             scanf("%d", (pdatos+d));
28         }
29         OrdenarBurbuja(pdatos, nelem);
30         printf("\n Que elemento desea buscar?");
31         scanf("%d", &dbuscar);
32         result = BusquedaBinaria(pdatos, nelem, dbuscar);
33         if(result!=-1)
34             printf("\n %d Se encuentra en la posición %d del array.", dbuscar, result);
35         else
36             printf("\n %d no se encuentra en el array.\n", dbuscar);
37     }
38     int BusquedaBinaria(int lista[], int n, int clave){
39         int central, bajo, alto;
40         int valorCentral;
41         bajo = 0;
42         alto = n-1;
43         while (bajo <= alto){
44             central = (bajo+alto)/2;
45             valorCentral = lista[central];
46
47             if(clave == valorCentral)
48                 return(central);
49             else if (clave < valorCentral)
50                 alto = central-1;
51             else
52                 bajo = central+1;
53         }
54         return(-1);
55     }

```

Fig5. Código binario2 completo.

```

PS C:\Users\HP\Desktop\Pame\output> & .\BINARIO2.exe
Cuantos elementos desea en el array?11
Elemento[0]:11
Elemento[1]:9
Elemento[2]:8
Elemento[3]:10
Elemento[4]:6
Elemento[5]:7
Elemento[6]:4
Elemento[7]:5
Elemento[8]:2
Elemento[9]:1
Elemento[10]:3

Que elemento desea buscar?3

3 Se encuentra en la posición 2 del array.
PS C:\Users\HP\Desktop\Pame\output>

```

Fig6. Funcionamiento del código binario2.

B. Código de Git

<https://github.com/Pamela1023/git-ejemplo/tree/main/Primer%20Parcial>